WHAT IS CLAIMED IS:

1. A method of obtaining a scalar value from a vector register for use in a mixed vector and scalar instruction, comprising:

providing a vector in a vector register file; and

embedding a location identifier of the scalar value within the vector in bits comprising the mixed vector and scalar instruction.

2. The method of claim 1, further including defining the mixed scalar and vector instruction such that at least one dedicated position bit is provided in a bit format of the instruction which provides the location identifier of the scalar value within the vector.

3. The method of claim 2, wherein a dimension of the vector is 2^n, and further including providing n bits in the bit format for indicating the location of the scalar value within the vector.

4. The method of claim 1, further including embedding the location identifier in an op code provided in the instruction.

5. The method of claim 4, further including embedding the location identifier in a secondary op code provided in the instruction.

6. The method of claim 1, further including using the instruction in a data processor having a paired singles execution unit, wherein two single precision values constitute the vector.

7. A data processor, comprising a vector processing unit, a vector register file and an instruction set, wherein the instruction set includes at least one mixed vector and scalar instruction having a bit format in which a location of a scalar value within a vector needed to execute the instruction is embedded.

8. The data processor of claim 7, wherein the location of the scalar value is embedded in a secondary op code of the instruction.

9. The data processor of claim 7, wherein at least one dedicated bit is provided in the bit format of the instruction to provide the location of the scalar value within the vector.

10. The data processor of claim 9, wherein the vector has a dimension of 2^n and n dedicated bits are provided in the instruction to provide the location of the scalar value within the vector.

11. The data processor of claim 7, wherein the mixed scalar and vector instruction specifies vector registers for all operands needed to execute the instruction.

12. The data processor of claim 1, wherein the vector has a dimension of two.

13. The data processor of claim 2, wherein the vector processing unit is a paired singles unit which processes two single-precision floating point values in parallel.

14. An instruction for a data processor, comprising a bit format which includes bits designating a first source vector register, bits designating a second source vector register and bits which indicate a location of a scalar operand within a vector register for use in executing the instruction.

15. The instruction of claim 14, wherein the bits which indicate the location of the scalar value within the vector register indicate a location within one of the first source vector register and the second source vector register.

16. The instruction of claim 14, further including bits which encode a primary op code and bit which encode a secondary op code.

17. The instruction of claim 16, wherein the bits which indicate the location of the scalar value within the vector register are embedded in the bits comprising one of the primary op code and secondary op code.

18. The instruction of claim 14, wherein the instruction is executable on a microprocessor having a vector processing unit.

19. The instruction of claim 18, wherein the instruction is executable on a microprocessor having a vector processing unit in the form of a paired singles unit.

20. An information processor, including a decoder for decoding instructions including at least some graphics instructions and at least one paired singles instruction, wherein the decoder is operable to decode a 32-bit

paired singles floating point add instruction, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing a pair of 32-bit single-precision floating point values resulting from the paired singles floating point add instruction, bits 11-15 designate a floating point source register storing a pair of 32-bit single-precision floating point values, bits 16-20 designate a further floating point source register storing a pair of 32-bit single-precision floating point values, bits 21-25 encode a reserved field of "00000", bits 26-30 encode a secondary op code of 21, and bit 31 comprises a record bit indicating updating of a condition register.

21. An information processor, including a decoder for decoding instructions including at least some graphics instructions and at least one paired singles instruction, wherein the decoder is operable to decode a 32-bit paired-single-scalar-vector-multiply-add-high (ps_madds0x) instruction wherein a high order word of a paired singles register is used as a scalar, and further wherein the ps_madds0x instruction includes bits 0 through 31, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing the results of the instruction, bits 11-15 designate a first floating point register as a first source storing a first pair of 32-bit single-precision floating point values, bits 16-20 designate a second floating point register as a second source storing a second pair of 32-bit single-precision floating point values, bits 21-25 designate a third floating point register as a third source storing a third pair of 32-bit single-precision

floating point values, bits 26-30 encode a secondary op code of 14 and bit 31 comprises a record bit indicating updating of a condition register.

22. The information processor, including a decoder for decoding instructions including at least some graphics instructions and at least one paired singles instruction, wherein the decoder is operable to decode a 32-bit paired-single-scalar-vector-multiply-add-low (ps_madds1x) instruction wherein a low order word of a paired singles register is used as a scalar, and further wherein the ps_madds1x instruction includes bits 0 through 31, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing the results of the instruction, bits 11-15 designate a first floating point register as a first source storing a first pair of 32-bit single-precision floating point values, bits 16-20 designate a second floating point register as a second source storing a second pair of 32-bit single-precision floating point values, bits 21-25 designate a third floating point register as a third source storing a third pair of 32-bit single-precision floating point values, bits 26-30 encode a secondary op code of 15 and bit 31 comprises a record bit indicating updating of a condition register.

23. An information processor, including a decoder for decoding instructions including at least some graphics instructions and at least one paired singles instruction, wherein the decoder is operable to decode a special purpose register command bit pattern including a special purpose register bit encoding whether paired singles operation is enabled, wherein the special purpose register bit is the third bit in the bit pattern.

24. A decoder for decoding instructions including at least some graphics instructions, wherein the decoder is operable to decode:

a 32-bit paired singles floating point add instruction, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing a pair of 32-bit single-precision floating point values resulting from the paired singles floating point add instruction, bits 11-15 designate a floating point source register storing a pair of 32-bit single-precision floating point values, bits 16-20 designate a further floating point source register storing a pair of 32-bit single-precision floating point values, bits 21-25 encode a reserved field of "00000", bits 26-30 encode a secondary op code of 21, and bit 31 comprises a record bit indicating updating of a condition register;

a 32-bit paired-single-scalar-vector-multiply-add-high (ps_madds0x) instruction wherein a high order word of a paired singles register is used as a scalar, and further wherein the ps_madds0x instruction includes bits 0 through 31, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing the results of the instruction, bits 11-15 designate a first floating point register as a first source storing a first pair of 32-bit single-precision floating point values, bits 16-20 designate a second floating point register as a second source storing a second pair of 32-bit single-precision floating point values, bits 21-25 designate a third floating point register as a third source storing a third pair of 32-bit single-precision floating point values, bits 26-30 encode a secondary op code of 14 and bit 31 comprises a record bit indicating updating of a condition register; and

a 32-bit paired-single-scalar-vector-multiply-add-low (ps_madds1x) instruction wherein a low order word of a paired singles register is used as a scalar, and further wherein the ps_madds1x instruction includes bits 0 through 31, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing the results of the instruction, bits 11-15 designate a first floating point register as a first source storing a first pair of 32-bit single-precision floating point values, bits 16-20 designate a second floating point register as a second source storing a second pair of 32-bit single-precision floating point values, bits 21-25 designate a third floating point register as a third source storing a third pair of 32-bit single-precision floating point values, bits 26-30 encode a secondary op code of 15 and bit 31 comprises a record bit indicating updating of a condition register.

25. A storage medium storing a plurality of instructions including at least some graphics instructions and a 32-bit paired singles floating point add instruction, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing a pair of 32-bit single-precision floating point values resulting from the paired singles floating point add instruction, bits 11-15 designate a floating point source register storing a pair of 32-bit single-precision floating point values, bits 16-20 designate a further floating point source register storing a pair of 32-bit single-precision floating point values, bits 21-25 encode a reserved field of "00000", bits 26-30 encode a secondary op code of 21, and bit 31 comprises a record bit indicating updating of a condition register.

26. A storage medium storing a plurality of instructions including at least some graphics instructions and a 32-bit paired-single-scalar-vector-multiply-add-high (ps_madds0x) instruction wherein a high order word of a paired singles register is used as a scalar, and further wherein the ps_madds0x instruction includes bits 0 through 31, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing the results of the instruction, bits 11-15 designate a first floating point register as a first source storing a first pair of 32-bit single-precision floating point values, bits 16-20 designate a second floating point register as a second source storing a second pair of 32-bit single-precision floating point values, bits 21-25 designate a third floating point register as a third source storing a third pair of 32-bit single-precision floating point values, bits 26-30 encode a secondary op code of 14 and bit 31 comprises a record bit indicating updating of a condition register.

27. A storage medium storing a plurality of instructions including at least some graphics instructions and a 32-bit paired-single-scalar-vector-multiply-add-low (ps_madds1x) instruction wherein a low order word of a paired singles register is used as a scalar, and further wherein the ps_madds1x instruction includes bits 0 through 31, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing the results of the instruction, bits 11-15 designate a first floating point register as a first source storing a first pair of 32-bit single-precision floating point values, bits 16-20 designate a second floating point register as a second source storing a second pair of 32-bit single-precision floating point values,

bits 21-25 designate a third floating point register as a third source storing a third pair of 32-bit single-precision floating point values, bits 26-30 encode a secondary op code of 15 and bit 31 comprises a record bit indicating updating of a condition register.

28. A storage medium storing a plurality of instructions including at least some graphics instructions and:

a 32-bit paired singles floating point add instruction, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing a pair of 32-bit single-precision floating point values resulting from the paired singles floating point add instruction, bits 11-15 designate a floating point source register storing a pair of 32-bit single-precision floating point values, bits 16-20 designate a further floating point source register storing a pair of 32-bit single-precision floating point values, bits 21-25 encode a reserved field of "00000", bits 26-30 encode a secondary op code of 21, and bit 31 comprises a record bit indicating updating of a condition register;

a 32-bit paired-single-scalar-vector-multiply-add-high (ps_madds0x) instruction wherein a high order word of a paired singles register is used as a scalar, and further wherein the ps_madds0x instruction includes bits 0 through 31, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing the results of the instruction, bits 11-15 designate a first floating point register as a first source storing a first pair of 32-bit single-precision floating point values, bits 16-20 designate a second floating point register as a second source storing a second pair of 32-

bit single-precision floating point values, bits 21-25 designate a third floating point register as a third source storing a third pair of 32-bit single-precision floating point values, bits 26-30 encode a secondary op code of 14 and bit 31 comprises a record bit indicating updating of a condition register; and

a 32-bit paired-single-scalar-vector-multiply-add-low (ps_madds1x) instruction wherein a low order word of a paired singles register is used as a scalar, and further wherein the ps_madds1x instruction includes bits 0 through 31, wherein bits 0-5 encode a primary op code of 4, bits 6-10 designate a floating point destination register for storing the results of the instruction, bits 11-15 designate a first floating point register as a first source storing a first pair of 32-bit single-precision floating point values, bits 16-20 designate a second floating point register as a second source storing a second pair of 32-bit single-precision floating point values, bits 21-25 designate a third floating point register as a third source storing a third pair of 32-bit single-precision floating point values, bits 26-30 encode a secondary op code of 15 and bit 31 comprises a record bit indicating updating of a condition register.